H 5

```
FFFFFFFFFFFFFFF         111               111        XXX         XXX
FFFFFFFFFFFFFFF         111               111        XXX         XXX
FFFFFFFFFFFFFFF         111               111        XXX         XXX
FFF                  111111            111111        XXX         XXX
FFF                  111111            111111        XXX         XXX
FFF                  111111            111111        XXX         XXX
FFF                     111               111           XXX   XXX
FFF                     111               111           XXX   XXX
FFF                     111               111           XXX   XXX
FFFFFFFFFFFF            111               111              XXX
FFFFFFFFFFFF            111               111              XXX
FFFFFFFFFFFF            111               111              XXX
FFF                     111               111           XXX   XXX
FFF                     111               111           XXX   XXX
FFF                     111               111           XXX   XXX
FFF                     111               111        XXX         XXX
FFF                     111               111        XXX         XXX
FFF                     111               111        XXX         XXX
FFF                  111111111         111111111     XXX         XXX
FFF                  111111111         111111111     XXX         XXX
FFF                  111111111         111111111     XXX         XXX
```

```
_$25

Symb
----
IOC$
IO_C
IO_C
IO_D
IO_F
IO_S
KILL

KILL
KILL
LB_E
LB_D
LB_F
LB_M
LB_L
LOCA
LOCA
LOCK

LOCK
LOCK
LOCK
LOC_
LOC_
L_CO
L_CO
L_DA
L_DA
MAIN
MAKE
MAKE
MAKE
MAKE
MAKE
MAKE

MAKE
MAKE
MAP_
MAP_

MAP
MARK
MARK
MARK
MARK
MARK
```

FILEID**MAKNMB   N 7

```
MM      MM    AAAAAA    KK      KK  NN      NN  MM      MM  BBBBBBBB
MM      MM    AAAAAA    KK      KK  NN      NN  MM      MM  BBBBBBBB
MMMM  MMMM  AA      AA  KK      KK  NN      NN  MMMM  MMMM  BB      BB
MMMM  MMMM  AA      AA  KK      KK  NN      NN  MMMM  MMMM  BB      BB
MM  MM  MM  AA      AA  KK    KK    NNNN    NN  MM  MM  MM  BB      BB
MM  MM  MM  AA      AA  KK    KK    NNNN    NN  MM  MM  MM  BB      BB
MM      MM  AA      AA  KKKKKK      NN  NN  NN  MM      MM  BBBBBBBB
MM      MM  AA      AA  KKKKKK      NN  NN  NN  MM      MM  BBBBBBBB
MM      MM  AAAAAAAAAA  KK    KK    NN    NNNN  MM      MM  BB      BB
MM      MM  AAAAAAAAAA  KK    KK    NN    NNNN  MM      MM  BB      BB
MM      MM  AA      AA  KK      KK  NN      NN  MM      MM  BB      BB    ....
MM      MM  AA      AA  KK      KK  NN      NN  MM      MM  BB      BB    ....
MM      MM  AA      AA  KK      KK  NN      NN  MM      MM  BBBBBBBB      ....
MM      MM  AA      AA  KK      KK  NN      NN  MM      MM  BBBBBBBB      ....

LL            IIIIII      SSSSSSSS
LL            IIIIII      SSSSSSSS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II      SS
LL              II        SSSSSS
LL              II        SSSSSS
LL              II              SS
LL              II              SS
LL              II              SS
LL              II              SS
LLLLLLLLLL    IIIIII    SSSSSSSS
LLLLLLLLLL    IIIIII    SSSSSSSS
```

MAK
V04

; R

```
0 MODULE MAKNMB (
0                   LANGUAGE (BLISS32),
0                   IDENT = 'V04-000'
0                   ) =
1 BEGIN
1
1 !
1 !****************************************************************************
1 !*                                                                          *
1 !*  COPYRIGHT (c) 1978, 1980, 1982, 1984 BY                                 *
1 !*  DIGITAL EQUIPMENT CORPORATION, MAYNARD, MASSACHUSETTS.                  *
1 !*  ALL RIGHTS RESERVED.                                                    *
1 !*                                                                          *
1 !*  THIS SOFTWARE IS FURNISHED UNDER A LICENSE AND MAY BE USED AND COPIED   *
1 !*  ONLY IN  ACCORDANCE WITH  THE  TERMS  OF  SUCH  LICENSE  AND WITH THE   *
1 !*  INCLUSION OF THE ABOVE COPYRIGHT NOTICE. THIS SOFTWARE OR  ANY  OTHER   *
1 !*  COPIES THEREOF MAY NOT BE PROVIDED OR OTHERWISE MADE AVAILABLE TO ANY   *
1 !*  OTHER PERSON.  NO TITLE TO AND OWNERSHIP OF  THE  SOFTWARE IS  HEREBY   *
1 !*  TRANSFERRED.                                                            *
1 !*                                                                          *
1 !*  THE INFORMATION IN THIS SOFTWARE IS  SUBJECT TO CHANGE WITHOUT NOTICE   *
1 !*  AND  SHOULD  NOT  BE  CONSTRUED AS  A COMMITMENT BY DIGITAL EQUIPMENT   *
1 !*  CORPORATION.                                                            *
1 !*                                                                          *
1 !*  DIGITAL ASSUMES NO RESPONSIBILITY FOR THE USE  OR  RELIABILITY OF ITS   *
1 !*  SOFTWARE ON EQUIPMENT WHICH IS NOT SUPPLIED BY DIGITAL.                 *
1 !*                                                                          *
1 !*                                                                          *
1 !****************************************************************************
1
1 !++
1 !
1 ! FACILITY:  F11ACP Structure Level 1
1 !
1 ! ABSTRACT:
1 !
1 !       This routine converts a file name string into the
1 !       RAD-50 name block format.
1 !
1 ! ENVIRONMENT:
1 !
1 !       STARLET operating system, including privileged system services
1 !       and internal exec routines.
1 !
1 !--
1 !
1 !
1 ! AUTHOR:  Andrew C. Goldstein,  CREATION DATE:  2-Jan-1977  17:06
1 !
1 ! MODIFIED BY:
1 !
1 !       V03-004 CDS0003         Christian D. Saether    2-Jan-1984
1 !               Use longword addressing on external FIL$ routine.
1 !
1 !       V03-003 CDS0002         Christian D. Saether    6-Dec-1983
1 !               Change LIB$ references to FIL$.
1 !
```

```
1 !        V03-002 ACG0302          Andrew C. Goldstein,    3-Dec-1982  13:55
1 !                Add $ and _ to file names, allow long names
1 !
1 !        V03-001 CDS0001          C. Saether              1-Jul-1982
1 !                Don't force absolute addressing mode when declaring
1 !                external routine lib$cvt_dtb.
1 !
1 !        A0101   ACG0057          Andrew C. Goldstein,    10-Aug-1979  16:41
1 !                Wild card interface changes
1 !
1 !        A0100   ACG00001         Andrew C. Goldstein,  10-Oct-1978  20:03
1 !        Previous revision history moved to F11A.REV
1 !**
1
1
1 LIBRARY 'SYS$LIBRARY:LIB.L32';
1 REQUIRE 'SRC$:FCPDEF.B32';
1
1
1 ! Linkages to subroutines in this module.
1 !
1
1 LINKAGE
1        L_GETCHAR          = JSB :
1                             NOPRESERVE (5)
1                             GLOBAL (COUNT = 6, STRINGP = 7, FCOUNT = 8),
1
1        L_GETSTAR          = JSB :
1                             GLOBAL (COUNT = 6, STRINGP = 7),
1
1        L_TYPE             = JSB :
1                             GLOBAL (COUNT = 6, STRINGP = 7);
1
1 ! Routines in this module
1 !
1
1 FORWARD ROUTINE
1        MAKE_NAMEBLOCK     : NOVALUE,        ! main routine
1        GETCHAR            : L_GETCHAR,      ! get RAD-50 set character
1        GETSTAR            : L_GETSTAR,      ! get star character, if any
1        TYPE               : L_TYPE;         ! determine type of current character
```

```
GLOBAL ROUTINE MAKE_NAMEBLOCK (LENGTH, STRING, NAMEBLOCK) : NOVALUE =

!++
!
! FUNCTIONAL DESCRIPTION:
!
!       This routine converts a file name string into the
!       RAD-50 name block format.
!
! CALLING SEQUENCE:
!       MAKE_NAMEBLOCK (ARG1, ARG2, ARG3)
!
! INPUT PARAMETERS:
!       ARG1: length of file name string
!       ARG2: address of file name string
!
! IMPLICIT INPUTS:
!       NONE
!
! OUTPUT PARAMETERS:
!       ARG3: address of file name block
!
! IMPLICIT OUTPUTS:
!       NONE
!
! ROUTINE VALUE:
!       NONE
!
! SIDE EFFECTS:
!       NONE
!
!--

BEGIN

MAP
        NAMEBLOCK       : REF BBLOCK;   ! name block arg

GLOBAL REGISTER
        COUNT   = 6,                    ! characters remaining in string
        STRINGP = 7     : REF VECTOR [,BYTE],  ! string pointer
        FCOUNT  = 8;                    ! count of chars in current field

LOCAL
        VERSION,                        ! file version number
        P,                              ! string scan pointer
        BLOCKP          : REF VECTOR [,WORD]; ! pointer into name block

EXTERNAL ROUTINE
        FIL$CVT_DTB     : ADDRESSING_MODE (GENERAL); ! decimal to binary convert

! Initialize all the locals.
!

CH$FILL (0, NMB$C_LENGTH, .NAMEBLOCK);  ! zero the entire block
STRINGP = .STRING;                      ! set up string pointer
P = CH$FIND_CH (.LENGTH, .STRINGP, ' '); ! look for a terminating space
```

```
2 COUNT = .P - .STRINGP;                     ! compute count
2 IF CH$FAIL (.P)
2 THEN COUNT = .LENGTH;                      ! use whole string if no space
2 BLOCKP = NAMEBLOCK[NMB$W_NAME];            ! point to name field in block
2 FCOUNT = 0;                                ! init chars in field count
2
2
2 ! Build the name field, consisting of 3 words of 3 RAD-50 characters per word.
2 !
2
2 DECR I FROM 3 TO 1 DO
3     BEGIN
3     DECR J FROM 3 TO 1 DO
3         BLOCKP[0] = .BLOCKP[0] * 40 + GETCHAR ();
3     BLOCKP = .BLOCKP + 2;
2     END;
2
2 ! Eat remaining trailing name field characters.
2 !
2
2 WHILE 1 DO
3     BEGIN
3     CASE TYPE () FROM 0 TO 6 OF
3         SET
3         [0,4,5,6]: EXITLOOP;
3         [INRANGE, OUTRANGE]:
4             BEGIN
4             COUNT = .COUNT - 1;
4             STRINGP = .STRINGP + 1;
3             END;
3         TES;
2     END;
2
2 IF GETSTAR ()                              ! set wild card bits if star
2 THEN
3     BEGIN
3     NAMEBLOCK[NMB$V_WILD] = 1;
3     NAMEBLOCK[NMB$V_ALLNAM] = 1;
2     END;
2
2 ! Pick up the name delimiter, which is either dot or end of string.
2 !
2
2 CASE TYPE () FROM 1 TO 5 OF
2     SET
2     [1,2,3,4]:  ERR_EXIT (SS$_BADFILENAME);
3     [5]:        BEGIN
3                 COUNT = .COUNT - 1;        ! pick up the character
3                 STRINGP = .STRINGP + 1;
2                 END;
2     [OUTRANGE]: 0;
2     TES;
2
2 ! Now build the type field, consisting of 1 word of 3 RAD-50 characters.
2 !
2
2 FCOUNT = 0;                                ! re-init chars in field count
```

```
DECR J FROM 3 TO 1 DO
    BLOCKP[0] = .BLOCKP[0] * 40 + GETCHAR ();

! Eat remaining trailing type field characters.
!

WHILE 1 DO
    BEGIN
    CASE TYPE () FROM 0 TO 6 OF
        SET
        [0,4,5,6]: EXITLOOP;
        [INRANGE, OUTRANGE]:
            BEGIN
            COUNT = .COUNT - 1;
            STRINGP = .STRINGP + 1;
            END;
        TES;
    END;

IF GETSTAR ()                               ! set wild card bits if star
THEN
    BEGIN
    NAMEBLOCK[NMB$V_WILD] = 1;
    NAMEBLOCK[NMB$V_ALLTYP] = 1;
    END;

! Pick up the type delimiter, which may be dot, semicolon, or end of string.
!

CASE TYPE () FROM 1 TO 6 OF
    SET
    [1,2,3,4]:  ERR_EXIT (SS$_BADFILENAME);
    [5,6]:      BEGIN
                COUNT = .COUNT - 1;     ! pick up the character
                STRINGP = .STRINGP + 1;
                END;
    [OUTRANGE]: 0;
    TES;

! If the version is not wild card and there are still characters present,
! get the binary version number.
!

IF GETSTAR ()                               ! set wild card bits if star
THEN
    BEGIN
    NAMEBLOCK[NMB$V_WILD] = 1;
    NAMEBLOCK[NMB$V_ALLVER] = 1;
    END
ELSE IF .COUNT GTR 0
THEN
    BEGIN
    BLOCKP = .BLOCKP + 2;
    IF NOT FIL$CVT_DTB (.COUNT, .STRINGP, VERSION)
    THEN ERR_EXIT (SS$_BADFILENAME);
    IF .VERSION GTRU 32767
```

```
MAKNMB                                                          G 8
V04-000                                                  16-Sep-1984 00:43:42    VAX-11 Bliss-32 V4.0-742                  Page   6
                                                         14-Sep-1984 12:30:34    DISK$VMSMASTER:[F11X.SRC]MAKNMB.B32;1      (2)

;  271        1260  3        THEN ERR_EXIT (SS$_BADFILEVER);
;  272        1261  3        (.BLOCKP)<0,16> = .VERSION;
;  273        1262  2        END;
;  274        1263  2
;  275        1264  2    RETURN 1;
;  276        1265  2
;  277        1266  1    END;                                            ! end of routine MAKE_NAMEBLOCK


                                                                              .TITLE  MAKNMB
                                                                              .IDENT  \V04-000\

                                                                              .EXTRN  FIL$CVT_DTB

                                                                              .PSECT  $CODE$,NOWRT,2

                                                         0FFC 00000           .ENTRY  MAKE_NAMEBLOCK, Save R2,R3,R4,R5,R6,R7,R8,-  ; 1089
                                                                                      R9,R10,R11                                    :
                               5B       0000V   CF    9E 00002              MOVAB   GETSTAR, R11                                  :
                               5A       0000V   CF    9E 00007              MOVAB   TYPE, R10                                     :
                               5E               04    C2 0000C              SUBL2   #4, SP                                        :
      28          00           6E               00    2C 0000F              MOVC5   #0, (SP), #0, #40, @NAMEBLOCK                 : 1143
                                        0C      BC       00014                                                                    :
                               57       08      AC    D0 00016              MOVL    STRING, STRINGP                               : 1144
                  67      04   AC               20    3A 0001A              LOCC    #32, LENGTH, (STRINGP)                        : 1145
                                                02    12 0001F              BNEQ    1$                                            :
                                                51    D4 00021              CLRL    R1                                            :
                  56           51               57    C3 00023 1$:          SUBL3   STRINGP, P, COUNT                             : 1146
                                                51    D5 00027              TSTL    P                                             : 1147
                                                04    12 00029              BNEQ    2$                                            :
                               56       04      AC    D0 0002B              MOVL    LENGTH, COUNT                                 : 1148
                  52      0C   AC               06    C1 0002F 2$:          ADDL3   #6, NAMEBLOCK, BLOCKP                         : 1149
                                                58    D4 00034              CLRL    FCOUNT                                        : 1150
                               59               03    D0 00036              MOVL    #3, I                                         : 1156
                               53               03    D0 00039 3$:          MOVL    #3, J                                         : 1158
                               54               62    3C 0003C 4$:          MOVZWL  (BLOCKP), R4                                  : 1159
                               54               28    C4 0003F              MULL2   #40, R4                                       :
                                            0000V     30 00042              BSBW    GETCHAR                                       :
                  62           54               50    A1 00045              ADDW3   R0, R4, (BLOCKP)                              :
                               F0               53    F5 00049              SOBGTR  J, 4$                                         :
                               52               02    C0 0004C              ADDL2   #2, BLOCKP                                    : 1160
                               E7               59    F5 0004F              SOBGTR  I, 3$                                         : 1156
                                                6A    16 00052 5$:          JSB     TYPE                                          : 1168
                  06           00               50    CF 00054              CASEL   R0, #0, #6                                    :
  000E          000E         000E             0014       00058 6$:          .WORD   8$-6$,-                                       :
                0014         0014             0014       00060                      7$-6$,-                                       :
                                                                                    7$-6$,-                                       :
                                                                                    7$-6$,-                                       :
                                                                                    8$-6$,-                                       :
                                                                                    8$-6$,-                                       :
                                                                                    8$-6$                                         :
                                                56    D7 00066 7$:          DECL    COUNT                                         : 1173
                                                57    D6 00068              INCL    STRINGP                                       : 1174
                                                E6    11 0006A              BRB     5$                                            : 1168
                                                6B    16 0006C 8$:          JSB     GETSTAR                                       : 1179
                               10               50    E9 0006E              BLBC    R0, 9$                                        :
                               50       0C      AC    D0 00071              MOVL    NAMEBLOCK, R0                                 : 1182
```

```
MAKNMB                                          H 8
V04-000                                  16-Sep-1984 00:43:42    VAX-11 Bliss-32 V4.0-742                 Page   7
                                         14-Sep-1984 12:30:34    DISK$VMSMASTER:[F11X.SRC]MAKNMB.B32;1     (2)

                    11    A0          01  88 00075          BISB2   #1, 17(R0)                          :
                          50     0C   AC  D0 00079          MOVL    NAMEBLOCK, R0                       : 1183
                    10    A0          20  88 0007D          BISB2   #32, 16(R0)                         :
                                      6A  16 00081 9$:      JSB     TYPE                                : 1189
            04            01          50  CF 00083          CASEL   R0, #1, #4                          :
  0098    0098          0098        0098     00087 10$:     .WORD   23$-10$,-                           :
                                    000C     0008F                  23$-10$,-                           :
                                                                    23$-10$,-                           :
                                                                    23$-10$,-                           :
                                                                    11$-10$                             :
                                      04  11 00091          BRB     12$                                 :
                                      56  D7 00093 11$:     DECL    COUNT                               : 1193
                                      57  D6 00095          INCL    STRINGP                             : 1194
                                      58  D4 00097 12$:     CLRL    FCOUNT                              : 1202
                          53          03  D0 00099          MOVL    #3, J                               : 1204
                          54          62  3C 0009C 13$:     MOVZWL  (BLOCKP), R4                        : 1205
                          54          28  C4 0009F          MULL2   #40, R4                             :
                                   0000V  30 000A2          BSBW    GETCHAR                             :
            62            54          50  A1 000A5          ADDW3   R0, R4, (BLOCKP)                    :
                          F0          53  F5 000A9          SOBGTR  J, 13$                              :
                                      6A  16 000AC 14$:     JSB     TYPE                                : 1212
            06            00          50  CF 000AE          CASEL   R0, #0, #6                          :
  000E    000E          000E        0014     000B2 15$:     .WORD   17$-15$,-                           :
          0014          0014        0014     000BA                  16$-15$,-                           :
                                                                    16$-15$,-                           :
                                                                    16$-15$,-                           :
                                                                    17$-15$,-                           :
                                                                    17$-15$,-                           :
                                                                    17$-15$                             :
                                      56  D7 000C0 16$:     DECL    COUNT                               : 1217
                                      57  D6 000C2          INCL    STRINGP                             : 1218
                                      E6  11 000C4          BRB     14$                                 : 1212
                                      6B  16 000C6 17$:     JSB     GETSTAR                             : 1223
                          10          50  E9 000C8          BLBC    R0, 18$                             :
                          50     0C   AC  D0 000CB          MOVL    NAMEBLOCK, R0                       : 1226
                    11    A0          01  88 000CF          BISB2   #1, 17(R0)                          :
                          50     0C   AC  D0 000D3          MOVL    NAMEBLOCK, R0                       : 1227
                    10    A0          10  88 000D7          BISB2   #16, 16(R0)                         :
                                      6A  16 000DB 18$:     JSB     TYPE                                : 1233
            05            01          50  CF 000DD          CASEL   R0, #1, #5                          :
  003E    003E          003E        003E     000E1 19$:     .WORD   23$-19$,-                           :
                        000E        000E     000E9                  23$-19$,-                           :
                                                                    23$-19$,-                           :
                                                                    23$-19$,-                           :
                                                                    20$-19$,-                           :
                                                                    20$-19$                             :
                                      04  11 000ED          BRB     21$                                 :
                                      56  D7 000EF 20$:     DECL    COUNT                               : 1237
                                      57  D6 000F1          INCL    STRINGP                             : 1238
                                      6B  16 000F3 21$:     JSB     GETSTAR                             : 1247
                          11          50  E9 000F5          BLBC    R0, 22$                             :
                          50     0C   AC  D0 000F8          MOVL    NAMEBLOCK, R0                       : 1250
                    11    A0          01  88 000FC          BISB2   #1, 17(R0)                          :
                          50     0C   AC  D0 00100          MOVL    NAMEBLOCK, R0                       : 1251
                    10    A0          08  88 00104          BISB2   #8, 16(R0)                          :
                                          04 00108          RET                                         : 1247
                                      56  D5 00109 22$:     TSTL    COUNT                               : 1253
```

```
MAKNMB                                        I 8
V04-000                                 16-Sep-1984 00:43:42    VAX-11 Bliss-32 V4.0-742                  Page   8
                                        14-Sep-1984 12:30:34    DISK$VMSMASTER:[F11X.SRC]MAKNMB.B32;1      (2)

                                         28 15 0010B          BLEQ    26$                                ;
                         52              02 C0 0010D          ADDL2   #2, BLOCKP                         ; 1256
                                         5E DD 00110          PUSHL   SP                                 ; 1257
                         7E              56 7D 00112          MOVQ    COUNT, -(SP)                       ;
              00000000G  00              03 FB 00115          CALLS   #3, FIL$CVT_DTB                    ;
                         05              50 E8 0011C          BLBS    R0, 24$                            ;
                                 0818    8F BF 0011F  23$:    CHMU    #2072                              ; 1258
                                            04 00123          RET                                        ;
              00007FFF   8F              6E D1 00124  24$:    CMPL    VERSION, #32767                    ; 1259
                                         05 1B 0012B          BLEQU   25$                                ;
                                 0820    8F BF 0012D          CHMU    #2080                              ; 1260
                                            04 00131          RET                                        ;
                         62              6E B0 00132  25$:    MOVW    VERSION, (BLOCKP)                  ; 1261
                                            04 00135  26$:    RET                                        ; 1266

; Routine Size:  310 bytes,    Routine Base:  $CODE$ + 0000
```

```
1 ROUTINE GETCHAR : L_GETCHAR =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine returns the RAD-50 code of the next character in the
1 !       input string if it is in the RAD-50 set. If it is not, or end of
1 !       string has been reached, it returns zero.
1 !
1 ! CALLING SEQUENCE:
1 !       GETCHAR ()
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !       COUNT: characters remaining in string
1 !       STRINGP: string pointer
1 !       FCOUNT: chars in current field
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       character code
1 !
1 ! SIDE EFFECTS:
1 !       COUNT decremented and STRINGP advanced if legal character.
1 !
1 !--
1
2 BEGIN
2
2 REGISTER
2       CHAR            = 5;                    ! character in process
2
2 EXTERNAL REGISTER
2       COUNT   = 6,                            ! characters remaining in string
2       STRINGP = 7     : REF VECTOR [,BYTE],   ! string pointer
2       FCOUNT  = 8;                            ! count of chars in current field
2
2
2 ! Get the next character from the string and dispatch in its type.
2 !
2
2 CHAR = .STRINGP[0];
2
2 CASE TYPE () FROM 0 TO 8 OF
2     SET
2     [0,5,6]:                                  ! end, dot, or semicolon
2       CHAR = 0;
2     [1]:                                      ! upper case alpha
3       BEGIN
```

```
;  336          1324 3              CHAR = .CHAR - 'A' + 1;              ! convert to RAD-50 code
;  337          1325 3              COUNT = .COUNT - 1;                  ! advance to next character
;  338          1326 3              STRINGP = .STRINGP + 1;
;  339          1327 3              FCOUNT = .FCOUNT + 1;                ! count character in field
;  340          1328 2              END;
;  341          1329 2          [2]:                                     ! lower case alpha
;  342          1330 3              BEGIN
;  343          1331 3              CHAR = .CHAR - 'a' + 1;              ! convert to RAD-50 code
;  344          1332 3              COUNT = .COUNT - 1;                  ! advance to next character
;  345          1333 3              STRINGP = .STRINGP + 1;
;  346          1334 3              FCOUNT = .FCOUNT + 1;                ! count character in field
;  347          1335 2              END;
;  348          1336 2          [3]:                                     ! numeric
;  349          1337 3              BEGIN
;  350          1338 3              CHAR = .CHAR - '0' + 30;             ! convert to RAD-50 code
;  351          1339 3              COUNT = .COUNT - 1;                  ! advance to next character
;  352          1340 3              STRINGP = .STRINGP + 1;
;  353          1341 3              FCOUNT = .FCOUNT + 1;                ! count character in field
;  354          1342 2              END;
;  355          1343 2          [7]:                                     ! dollar sign
;  356          1344 3              BEGIN
;  357          1345 3              CHAR = 27;                           ! convert to RAD-50 code
;  358          1346 3              COUNT = .COUNT - 1;                  ! advance to next character
;  359          1347 3              STRINGP = .STRINGP + 1;
;  360          1348 3              FCOUNT = .FCOUNT + 1;                ! count character in field
;  361          1349 2              END;
;  362          1350 2          [8]:                                     ! underscore
;  363          1351 3              BEGIN
;  364          1352 3              CHAR = 29;                           ! convert to RAD-50 code
;  365          1353 3              COUNT = .COUNT - 1;                  ! advance to next character
;  366          1354 3              STRINGP = .STRINGP + 1;
;  367          1355 3              FCOUNT = .FCOUNT + 1;                ! count character in field
;  368          1356 2              END;
;  369          1357 2          [4]:                                     ! star - legal as only char in field
;  370          1358 3              BEGIN
;  371          1359 3              CHAR = 0;
;  372          1360 3              IF .FCOUNT NEQ 0 THEN ERR_EXIT (SS$_BADFILENAME);
;  373          1361 2              END;
;  374          1362 2          TES;
;  375          1363 2
;  376          1364 2      RETURN .CHAR;
;  377          1365 2
;  378          1366 1      END;                                         ! end of routine GETCHAR



                              55          67  9A 00000 GETCHAR:MOVZBL  (STRINGP), CHAR                          ; 1316
                                       0000V  30 00003         BSBW    TYPE                                     ; 1318
                   08         00          50  CF 00006         CASEL   R0, #0, #8                               ;
    0022         001C       0016        0012     0000A 1$:     .WORD   2$-1$,-                                  ;
    0027         0012       0012        0037     00012                 3$-1$,-                                  ;
                                        002C     0001A                 4$-1$,-                                  ;
                                                                       5$-1$,-                                  ;
                                                                       9$-1$,-                                  ;
                                                                       2$-1$,-                                  ;
```

```
                                                        2$-1$,-                                 ;
                                                        6$-1$,-                                 ;
                                                        7$-1$                                   ;
                          55  D4 0001C 2$:     CLRL     CHAR                                    ; 1321
                          2C  11 0001E         BRB      10$                                     ;
          55         C0   A5  9E 00020 3$:     MOVAB    -64(R5), CHAR                           ; 1324
                          13  11 00024         BRB      8$                                      ; 1325
          55         A0   A5  9E 00026 4$:     MOVAB    -96(R5), CHAR                           ; 1331
                          0D  11 0002A         BRB      8$                                      ; 1332
          55              12  C2 0002C 5$:     SUBL2    #18, CHAR                               ; 1338
                          08  11 0002F         BRB      8$                                      ; 1339
          55              1B  D0 00031 6$:     MOVL     #27, CHAR                               ; 1345
                          03  11 00034         BRB      8$                                      ; 1346
          55              1D  D0 00036 7$:     MOVL     #29, CHAR                               ; 1352
                          56  D7 00039 8$:     DECL     COUNT                                   ; 1353
                          57  D6 0003B         INCL     STRINGP                                 ; 1354
                          58  D6 0003D         INCL     FCOUNT                                  ; 1355
                          0B  11 0003F         BRB      10$                                     ; 1318
                          55  D4 00041 9$:     CLRL     CHAR                                    ; 1359
                          58  D5 00043         TSTL     FCOUNT                                  ; 1360
                          05  13 00045         BEQL     10$                                     ;
                   0818   8F  BF 00047         CHMU     #2072                                   ;
                              05 0004B         RSB                                              ;
          50              55  D0 0004C 10$:    MOVL     CHAR, R0                                ; 1364
                              05 0004F         RSB                                              ; 1366

; Routine Size:  80 bytes,    Routine Base:  $CODE$ + 0136
;  380     1367  1 ROUTINE GETSTAR : L_GETSTAR =
;  381     1368  1
;  382     1369  1 !++
;  383     1370  1 !
;  384     1371  1 ! FUNCTIONAL DESCRIPTION:
;  385     1372  1 !
;  386     1373  1 !       This routine gobbles the next character in the input string
;  387     1374  1 !       if it is a star.
;  388     1375  1 !
;  389     1376  1 ! CALLING SEQUENCE:
;  390     1377  1 !       GETSTAR ()
;  391     1378  1 !
;  392     1379  1 ! INPUT PARAMETERS:
;  393     1380  1 !       NONE
;  394     1381  1 !
;  395     1382  1 ! IMPLICIT INPUTS:
;  396     1383  1 !       COUNT: number of characters in input string
;  397     1384  1 !       STRINGP: input string pointer
;  398     1385  1 !
;  399     1386  1 ! OUTPUT PARAMETERS:
;  400     1387  1 !       NONE
;  401     1388  1 !
;  402     1389  1 ! IMPLICIT OUTPUTS:
;  403     1390  1 !       NONE
;  404     1391  1 !
;  405     1392  1 ! ROUTINE VALUE:
;  406     1393  1 !       1 if character was a star
;  407     1394  1 !       0 otherwise
;  408     1395  1 !
;  409     1396  1 ! SIDE EFFECTS:
;  410     1397  1 !       COUNT decremented, STRINGP incremented if character was star.
;  411     1398  1 !
;  412     1399  1 !--
;  413     1400  1
;  414     1401  2 BEGIN
;  415     1402  2
;  416     1403  2 EXTERNAL REGISTER
;  417     1404  2         COUNT   = 6,                    ! characters remaining in string
;  418     1405  2         STRINGP = 7     : REF VECTOR [,BYTE];   ! string pointer
;  419     1406  2
;  420     1407  2 IF .COUNT GTR 0 AND .STRINGP[0] EQL '*'
;  421     1408  2 THEN
;  422     1409  3     BEGIN
;  423     1410  3     COUNT = .COUNT - 1;
;  424     1411  3     STRINGP = .STRINGP + 1;
;  425     1412  3     1
;  426     1413  3     END
;  427     1414  2 ELSE
;  428     1415  2     0
;  429     1416  2
;  430     1417  1 END;                                    ! end of routine GETSTAR


                                     56  D5 00000 GETSTAR:TSTL     COUNT                              ; 1407
```

```
                              0D  15 00002         BLEQ    1$                          ;
                        2A    67  91 00004         CMPB    (STRINGP), #42              ;
                              08  12 00007         BNEQ    1$                          ;
                              56  D7 00009         DECL    COUNT                       ; 1410
                              57  D6 0000B         INCL    STRINGP                     ; 1411
                        50    01  D0 0000D         MOVL    #1, R0                      ; 1409
                                  05 00010         RSB                                 ;
                              50  D4 00011  1$:    CLRL    R0                          ; 1407
                                  05 00013         RSB                                 ; 1417

; Routine Size:  20 bytes,    Routine Base:  $CODE$ + 0186
```

**F

```
1 ROUTINE TYPE : L_TYPE =
1
1 !++
1 !
1 ! FUNCTIONAL DESCRIPTION:
1 !
1 !       This routine determines the type code of the current character
1 !       in the string.
1 !
1 ! CALLING SEQUENCE:
1 !       TYPE ()
1 !
1 ! INPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT INPUTS:
1 !       COUNT: number of characters left in string
1 !       STRINGP: string pointer
1 !
1 ! OUTPUT PARAMETERS:
1 !       NONE
1 !
1 ! IMPLICIT OUTPUTS:
1 !       NONE
1 !
1 ! ROUTINE VALUE:
1 !       type code of character:
1 !       0: end of string or non-RAD-50
1 !       1: upper case alpha
1 !       2: lower case alpha
1 !       3: numeric
1 !       4: star
1 !       5: dot
1 !       6: semicolon
1 !       7: $
1 !       8: _
1 !
1 ! SIDE EFFECTS:
1 !       NONE
1 !
1 !--
1
2 BEGIN
2
2 EXTERNAL REGISTER
2       COUNT   = 6,            ! characters remaining in string
2       STRINGP = 7     : REF VECTOR [,BYTE]; ! string pointer
2
2 ! Character match tables. First is low character of range, second is
2 ! high character. Type is table index of the matching range.
2 !
2
2 BIND
2       LOWCHAR         = UPLIT BYTE (0, 'Aa0*.;$_') : VECTOR [,BYTE],
2       HIGHCHAR        = UPLIT BYTE (0, 'Zz9* ;$_') : VECTOR [,BYTE];
2
2 ! If the string is empty return 0 as the type. Else search the tables.
```

```
;  489          1475  2 !
;  490          1476  2
;  491          1477  2 IF .COUNT LEQ 0 THEN RETURN 0;
;  492          1478  2
;  493          1479  2 INCR I FROM 1 TO 8 DO
;  494          1480  2     IF .STRINGP[0] GEQU .LOWCHAR[.I]
;  495          1481  2     AND .STRINGP[0] LEQU .HIGHCHAR[.I]
;  496          1482  2     THEN RETURN .I;
;  497          1483  2
;  498          1484  2 ERR_EXIT (SS$_BADFILENAME);                ! other characters are illegal
;  499          1485  2
;  500          1486  2
;  501          1487  1 END;                                       ! end of routine TYPE


                                          00  0019A P.AAA:   .BYTE    0                          ;
          5F  24  3B  2E  2A  30  61  41  0019B             .ASCII   \Aa0*.;$_\                 ;
                                          00  001A3 P.AAB:   .BYTE    0                          ;
          5F  24  3B  2E  2A  39  7A  5A  001A4             .ASCII   \Zz9*.;$_\                 ;

                                                    LOWCHAR=          P.AAA
                                                    HIGHCHAR=         P.AAB


                                      56  D5 00000 TYPE:    TSTL     COUNT                              ; 1477
                                      1A  15 00002          BLEQ     3$                                 ;
                         50           01  D0 00004          MOVL     #1, I                              ; 1480
                    E2 AF40           67  91 00007 1$:      CMPB     (STRINGP), LOWCHAR[I]              ;
                                      07  1F 0000C          BLSSU    2$                                 ;
                    E4 AF40           67  91 0000E          CMPB     (STRINGP), HIGHCHAR[I]             ; 1481
                                      0B  1B 00013          BLEQU    4$                                 ;
               EE        50           08  F3 00015 2$:      AOBLEQ   #8, I, 1$                          ; 1480
                                0818  8F  BF 00019          CHMU     #2072                              ; 1484
                                          05 0001D          RSB                                         ;
                                      50  D4 0001E 3$:      CLRL     R0                                 ; 1487
                                          05 00020 4$:      RSB                                         ;

; Routine Size:  33 bytes,    Routine Base:  $CODE$ + 01AC


;  502          1488  1
;  503          1489  1 END
;  504          1490  0 ELUDOM




                                 PSECT SUMMARY

;      Name                    Bytes                          Attributes

;  $CODE$                             461  NOVEC,NOWRT,  RD ,  EXE,NOSHR,  LCL,  REL,  CON,NOPIC,ALIGN(2)
;                              Library Statistics
;
;                                              -------- Symbols --------       Pages       Processing
;       File                                   Total    Loaded   Percent       Mapped      Time
;
;    _$255$DUA28:[SYSLIB]LIB.L32;1             18619       25         0        1000         00:02.0




;                                      COMMAND QUALIFIERS

;       BLISS/CHECK=(FIELD,INITIAL,OPTIMIZE)/LIS=LIS$:MAKNMB/OBJ=OBJ$:MAKNMB MSRC$:MAKNMB/UPDATE=(ENH$:MAKNMB)

; Size:          443 code + 18 data bytes
; Run Time:         00:15.7
; Elapsed Time:     00:33.1
; Lines/CPU Min:      5694
; Lexemes/CPU-Min: 21183
; Memory Used:  176 pages
; Compilation Complete
```

0171 AH-BT13A-SE
VAX/VMS V4.0

DIGITAL EQUIPMENT CORPORATION
CONFIDENTIAL AND PROPRIETARY

PMS
LIS

QUOTAUTIL
LIS

MAKPTR
LIS

MATCHNAME
LIS

MPWIND
LIS

LOCKDB
LIS

LOCKERS
LIS

MAKACC
LIS

PARSNM
LIS

MAPVBN
LIS

IODONE
LIS

LOCKDN
LIS

MODIFY
LIS

MOUNT
LIS

NXTHDR
LIS

MAKNMB
LIS

MAKSTR
LIS